Next, in step 83, the invention creates a DD for the XML format as defined in Figure 5B. In step 84, the user can manually enter additional DDs. The user can now enter definitions for the rules that manipulate the data in the MRTs and DDs. Enter definitions for functions in step 85, for filters in step for 86, and for sorts in step 87. The methods used for defining these definitions are similar to those typically used in the art.

In FIG. 4B, an overview of the preferred method for managing the import and processing of new XML documents is described. In step 89, a user imports an XML document of a format previously imported into the invention. In step 90, the invention creates the PRIs and Management Record Pointer Instances (MRPIs) as described in FIG. 6 below. The preferred method, in step 91, next executes MRT pre-function filters on the fetched data, as discussed below with reference to FIG. 8. Then in step 92, the MRT functions are executed using the fetched and filtered data as described in FIG. 7 below. Next in step 93, MRT post-function filters are executed. These filters are executed with a similar process used in step 91. Now that the data from the data sources has been retrieved, the preferred method creates dynamic document pointers (DDP) for displaying the data in step 94 (See FIG. 9 and discussion below). As with MRTs, pre-function filters (See FIG. 18 and discussion below), functions (See FIG. 16 and discussion below), post-function filters, and sorts are executed in steps 95, 96, 97, and 98, respectively. Then in step 98, the dynamic document pointer's (DDP's) calculated fields, MRPF's calculated fields, and the PRIs are organized into panes and records for display in a window or for writing of XML document for each DD instance.

In FIG. 4C, an overview of the preferred method for managing the processing of data changes entered by users is described. The user can now enter revisions to update the data using the DD displays in step 100. In step 101, the method first determines whether the change includes a new PRI record because the user entered new unique identifier values for the affected PRT. If change is not a new PRI, step 102 updates the existing affected PRI. If a new PRI is required step 103 created the new PRI, and step 104 determines if the PRIs PRT is the lead PRT for a Management Record Pointer Family. If it is not a lead PRT, step 105 updates the appropriate MPRI to now include the new PRI. While if it is a new lead PRT, step 106 creates the new MRPI. Next, in step 107, the present invention executes the functions and filters for the revision. If any of the functions changes data values, step 108 sends those revisions to step 101 to be processed in the

same manner as user entered revisions. If the functions produced no data revisions, step 109 follows where the DDs 20 are updated to reflect the revision. Then in step 110, the method updates DDP records for display in a window or writing of an XML document.

In FIG. 4D, an overview of the preferred method for managing user changes to processing rules is described. In step 111, the user enters a change to one of the rules that manage the data. In step 112 the preferred method tests whether the revision is a function revision. If not, the process continues to step 114. However, if the revision is a function revision, the function is executed with the revision in step 113, and as described below with reference to FIG. 13. In step 114, the preferred method determines whether the revision is a filter revision. If not, the process continues to step 116. However, if the revision is a filter revision, the filter is re-executed using the revision in step 115, and as described below with reference to FIG. 14. Then in step 116, the preferred method test whether the revision is a sort revision. If not, the method continues to step 118. However, if the revision is a sort revision, the sort is executed in step 117. In step 118, the preferred method tests if processing the revision as described above resulted in revisions to data. If there are, the method loops to step 101. If the revision did not generate other revisions, the method moves to step 119 where it updates the display of the DDs to reflect changes to the sequence of data and the inclusion or omission of data affected by the changes to the filter criteria.

In FIG. 5, the preferred method for generating MRTs and their and their corresponding Management Record Pointer Families (MRPFs) is described. The process begins in step 120 where the user imports or enters an XML format. In step 121 where the invention identifies all the PRTs that comprise the XML document format. For every component within the XML document format that contains one or more components, the invention identifies a potential PRT. In step 122, the user indicates if the PRT already exists. If it does exist, in step 124 the user maps the field level components in the XML document to the fields in an existing PRT. The method then proceeds to step 129. If it does not exist the method automatically creates a new PRT. Step 126 checks the PRTs to determine if it is a multiple occurrence component: a multiple occurrence component can have more than one instance in a single XML document. If it is a multiple occurrence PRT, step 127 it creates a new MRPF with that PRT as its lead PRT (item 50 in FIG. 3). If it is a single occurrence PRT, step 128, the invention assigns the PRT to the MRPF where its parent PRT is the lead PRT. In

step 129, the method determines if the XML format has any remaining PRTs to process. If any remain the method proceeds to step 121 to process the next PRT.

In FIG. 6, the preferred method for copying the data in XML documents into the invention's DDs, MRTs, and PRTs. This process begins when the method receives a new XML document the user queried or some other program sent to the invention. In step 130, the XML document is read. In step 131 the components within the XML document that comprise one of its PRIs are read. Step 132 determines if the read components are for a new or existing PRI. If the PRI already exists, step 133, updates the existing PRI to reflect the new data values and step 134 updates the MRPF instance records (MRPIs) that use the changed PRI as described in FIG. 11. If the XML document includes a new PRI, step 135 creates it and step 136 updates the MRPIs that use the new PRI as described in FIG. 10. Step 137 determines if any additional PRIs exist within the XML document. If any remain, the invention returns to step 131 to process the next PRI. When all the PRIs are processed, step 138 recreates the DDs as described in FIG. 11.

Referring now to FIGS. 7A and 7B, the preferred method for executing functions will be described. A function is one or more mathematical, financial, and statistical calculations performed using the fields of the MRT's PRTs. Examples of functions include: add, subtract, multiply, divide, round, test, join, summarize, generate a random number, log, remainder, and calculate present value. Examples of "group by functions" include: any occurrence, first occurrence, last occurrence, sum, average, count, maximum, minimum, standard deviation, variance, and net present value. The present invention is particularly advantageous because any combination of PRT fields and any MRPF calculated fields can be used in a function. If the field being used is not in the pointer family or one of the fore parents of the pointer family to which the results are assigned, the present invention requests that the user enter the "group by function" it should perform on that field prior to using it in the calculation. For example, if a user wishes to calculate an order discount by multiplying the order line dollar amounts by the order header discount and then store the results in the order header, the user would specify the "sum" group by function for the order line dollar amounts because order lines are children not fore parents of the order header where the result is stored.